

Secure information embedding into 1D biomedical signals based on SPIHT



Óscar J. Rubio*, Álvaro Alesanco, José García

Communication Technologies Group, Aragón Institute of Engineering Research, University of Zaragoza, Edif. Ada Byron, C/María de Luna 3, 50018 Zaragoza, Spain

ARTICLE INFO

Article history:

Received 27 September 2012

Accepted 9 May 2013

Available online 23 May 2013

Keywords:

Biomedical signals

Distortion

E-health

Metadata embedding

Security

SPIHT

ABSTRACT

This paper proposes an encoding system for 1D biomedical signals that allows embedding metadata and provides security and privacy. The design is based on the analysis of requirements for secure and efficient storage, transmission and access to medical tests in e-health environment. This approach uses the 1D SPIHT algorithm to compress 1D biomedical signals with clinical quality, metadata embedding in the compressed domain to avoid extra distortion, digital signature to implement security and attribute-level encryption to support Role-Based Access Control. The implementation has been extensively tested using standard electrocardiogram and electroencephalogram databases (MIT-BIH Arrhythmia, MIT-BIH Compression and SCCN-EEG), demonstrating high embedding capacity (e.g. 3 KB in resting ECGs, 200 KB in stress tests, 30 MB in ambulatory ECGs), short delays (2–3.3 s in real-time transmission) and compression of the signal (by ≈ 3 in real-time transmission, by ≈ 5 in offline operation) despite of the embedding of security elements and metadata to enable e-health services.

© 2013 Elsevier Inc. All rights reserved.

1. Introduction

Medical tests need to evolve in the context of e-health [1] towards the new patient-centric paradigm of healthcare [2], which is supported by the Information and Communication Technologies (ICT). The core of these tests are biomedical signals (e.g. an electrocardiogram or an electroencephalogram), whose clinical meaning need to be complemented with additional information (annotations about the signal, personal data of the patient, his/her health status, allergies, medication) to enable his/her early diagnosis, continuous follow-up and customized care. To do so, and to guarantee adequate data availability (comprising secure and efficient storage, exchange and access), a method combining optimal encoding and protection of medical tests must be investigated. The resulting protected tests may facilitate the development of new secure ICT-based health services or the upgrade of existing (e.g. home monitoring, decision support systems, wearable sensor networks, e-prescribing [3]).

The requirements that digital medical tests must fulfill to fit the new e-health paradigm may be summarized as:

- *Information associated to the signal.* Without appropriate data, identifying the signal and enabling its interpreta-

tion, medical tests may become useless. For this reason the information in medical tests must be arranged as metadata using some data structure and bound to the signal to difficult its lost.

- *Signal compression.* Algorithms for signal compression remove redundancies contained by signals at different levels. These algorithms can be divided into two main categories: lossless, which retrieve the original signal; and lossy, which reach higher compression ratios than lossless at the cost of decreasing signal fidelity. The latter are more interesting since they permit saving much more bandwidth in transmission and disk space in storage. Nevertheless, in clinical applications the compression ratio must be limited by measurable quality parameters to hold the clinical meaning of the signal and avoid changing its diagnostic interpretation. Among lossy methods, there are three modalities [4]: direct methods (basing their detection of redundancies on direct analysis of the actual signal samples), transformation methods (mainly utilizing spectral and energy distribution analysis for detecting redundancies) and parameter extraction techniques (e.g. measurement of the probability distribution, subsequently utilized for classification based on *a priori* knowledge of the signal features). The second modality (e.g. discrete cosine transform [5], Karhunen-Loève transform [6], wavelets [7], etc.) generally yields better results, especially the wavelets, which provide a time–frequency representation

* Corresponding author. Tel.: +34 976762698; fax: +34 976762111.

E-mail addresses: orubio@unizar.es (Ó.J. Rubio), alesanco@unizar.es (Á. Alesanco), jjgarmo@unizar.es (J. García).

of the signal with varying resolution for fine description in both domains. Furthermore, the wavelet coefficients can also be compressed by exploiting their similarity, as the SPIHT algorithm does [8], in order to increase the final compression ratio.

- **Security and privacy** in storage and during transmission. Current legal regulations (the HIPAA [9], the PIPEDA [10], the LOPD [11], the Digital Signature Laws in several countries) demand that any personal health information must be protected, using adequate cryptographic means. The basic requirements are (1) encrypting all private data, (2) embedding a digital signature to verify data integrity and authenticate the signatory, and (3) encrypting the communications. Steganography may be used as a complement to introduce security or secret elements silently. Watermarking, marking all objects in the same way (e.g. to demonstrate ownership) and fingerprinting, marking each object specifically (e.g. to identify legitimate users) are the most typical applications of steganography.
- **Role-Based Access Control.** E-health services operate in scenarios with a variety of different stakeholders: patients, relatives, paramedics, nurses, primary care doctors/general practitioners, surgeons, medical specialists and subspecialists, teachers and medical students, researchers, laboratories, insurance companies, governmental oversight agencies, and non-governmental oversight. For the same patient, the information that each user is allowed to access must depend on his role: e.g. if the patient has AIDS, the nurses and the paramedics need to know, but probably not the researchers using his/her medical tests. Attribute-level encryption and de-identification are effective ways to overcome this issue.
- **Low complexity encoding and short access time.** Since the current tendency is building portable medical devices and wearable sensors, which often mount low power processors, the algorithms for encoding and protection should be as simple as possible to not overload them with complex calculations and reduce demand on the battery. Besides, fast execution and transmission are requirements to maintain availability of the test at good levels and allow real-time services.

There are many publications approaching these requisites separately (see for example [12–14] for signal compression, [15,16] for data embedding into signals and [17–21] for signal security). However, the aim of this work is to find an encoding and access system harmonizing compression, embedding and security. It will be focused on the case of 1D signals to facilitate the evaluation, but it could also be extended to signals of higher dimension. The rest of the paper is organized as follows. Section 2 depicts our proposal, which is evaluated in Section 3 with standard signal databases. The system implementation built for the evaluation is presented in Section 4. Finally the results are analyzed in Section 5, extracting conclusions and discussing future lines of work.

2. Materials and methods

The outline of the new encoding and access architecture for 1D biomedical tests is represented in Fig. 1. To fulfill the requirements in Section 1, it implements adequate methods for signal compression and embedding of additional measures, data of the patient and elements to provide security and privacy. The chosen compressor (Section 2.1) is not only fast and efficient but it also facilitates the embedding process (Section 2.2), in which security items are introduced together with the data (Section 2.3) to limit access to it. Real-time encoding and access is feasible since the encoder works with length-adjustable signal blocks (and additional measures) to constrain the delays, producing encoded units called *signal registers*. Then, these are sent to the Picture Archive and

Communication System (PACS), whose admin checks their integrity and authenticity and accesses the private data of the patient. Next, the *signal registers* are stored according to the PACS data model (e.g. patient/study/series/instance like in DICOM [22]). When this test is requested to the PACS by a user, it will be sent preserving its encoding, which ensures that the user can access only those contents authorized by his/her professional role. Besides, the signal may also be reconstructed by any user knowing the compression method, even if the presence of the embedded data is ignored. Finally, users who know the encoding can always access the embedded additional measures and detect corruption of the signal or the contents.

The former experience of the authors with medical protocols [23–26] and ECG signals [27,28] and the opinion of three independent physicians has inspired this architecture.

2.1. Signal compression

Among the variety of general methods for biomedical signal compression, the combination of wavelets and SPIHT not only obtains good results, it is also simple and returns a bitframe which can be truncated at any point with progressive lossy to lossless quality. Another outstanding property is that large amounts of data can be embedded and retrieved from truncated SPIHT bitframes with simplicity and secrecy.

2.1.1. SPIHT overview

SPIHT was firstly presented in [8] as an efficient method for coding wavelet coefficients in image (2-D) compression. In [12] the algorithm was adapted to the one-dimensional (1-D) case and applied to ECG signals, revealing that it was very efficient in compression and in computation when compared with previous ECG compression methods. Besides the SPIHT algorithm accounted with several desirable properties: multiresolution scalability, progressive lossy to lossless coding, compatibility with lossless entropy coding, low complexity (use of simple operators), moderate memory usage and symmetric coding-decoding. These features motivated the later extension of the algorithm to the three-dimensional (3-D) [29] and four-dimensional (4-D) cases [30], and its successful VLSI implementation in silicon for ECG real-time compression in low-power applications [31].

The principles of the SPIHT algorithm are partial ordering of the transform coefficients by magnitude with a set partitioning sorting algorithm, ordered bit plane transmission and exploitation of self-similarity across different layers. By following these principles, the encoder always transmits the most significant bit to the decoder.

Basically the (1-D) algorithm uses a *temporal orientation tree* structure to define the temporal parent-offspring relations in the wavelet domain, across consecutive layers. The set partitioning rule creates subset of subband coefficients c_i , whose indices will be used together with the coefficient indices (referred to as points) to create and update three related lists: the list of insignificant points (LIP), the list of insignificant sets (LIS), and the list of significant points (LSP). The outline of the algorithm is as follows:

1. **Initialization.** Set the list of significant points (LSP) as empty. Set the roots of similarity trees in the list of insignificant points (LIP) and insignificant sets (LIS). Set the significance threshold 2^n with $n = \lfloor \log_2(\max_i |c_i|) \rfloor$.
2. **Sorting pass.** Using the set partitioning algorithm distribute the appropriate indices of the coefficients to the LIP, LIS, and LSP.
3. **Refinement pass.** For each entry in the LSP significant for higher n , send the n th most significant bit to the decoder.

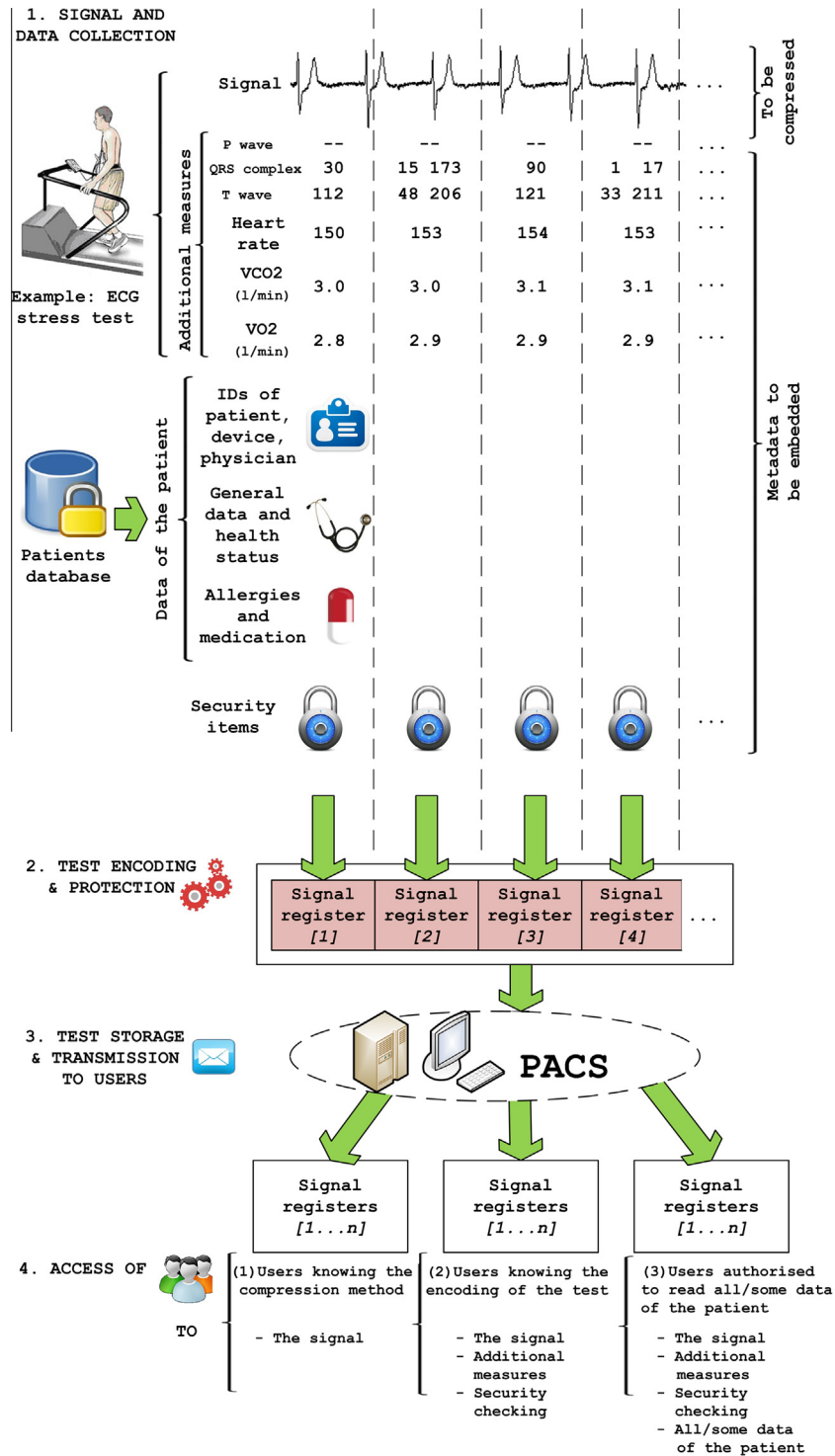


Fig. 1. Scheme for the encoding and access to 1D biomedical tests in e-health environments. Signal registers are depicted in Fig. 2.

- Decrement n by one and return to step 2 until the specified bitrate or distortion is reached.

A more detailed explanation of the algorithm can be found in [12].

2.1.2. Compression and distortion in SPIHT

Before compressing, the original signal is divided into contiguous non-overlapping blocks to boost the compression, reduce memory demands and allow real-time services. Adequate block

length values, balancing bandwidth requirements and delays, are discussed in Section 3.3.

Bounding distortion using SPIHT is a simple task if we use the fact that the Euclidean norm, which is used to measure the error, is invariant to the wavelet transform (since it is a unitary transformation). Thus, guaranteeing reconstruction quality can be easily done by controlling the value of the coded coefficients and calculating some distortion measure to stop the coding process when the desired distortion level is reached [27,28]. This is detailed in Section 3.2.

2.2. Embedding metadata within the signal

The embedding and retrieval of metadata (e.g. data of the patient, additional measures of the test, security items) in the SPIHT domain presents great advantages: high capacity, very low complexity and controllable signal distortion. As illustrated in Fig. 2, the metadata bits are placed after the SPIHT bitframe (to minimize distortion). These bits are kept for signal reconstruction, providing a common access to the signal regardless whether the user knows the encoding method (thus, the presence of additional contents) or just the compression algorithm, as illustrated in Fig. 1. The change in the quality of the reconstructed signal due to this extension of the SPIHT bitframe is low or moderate, as it will be shown in Section 3.3. Besides when it decreases, the SPIHT bitframe is progressively extended with new SPIHT bits until this negative effect is balanced out. It is not necessary to reconstruct the signal in the time domain to update the distortion when extending the SPIHT. This is feasible in the transform domain since the wavelet is a unitary transformation, and practical because the wavelet coefficients were already calculated for compression.

2.3. Metadata encoding, protection and access

Organizing and protecting the metadata to be embedded within the *signal registers* (see Fig. 1) are basic requirements to guarantee that corruption of the signal or the metadata can be detected and that access to the latter is suitably controlled. Although the Cryptographic Message Syntax [32] (implemented by DICOM) provide the means to digitally sign, digest, authenticate or encrypt any digital content, it presents disadvantages that the *signal registers* must avoid: the syntax to protect each piece of data is not separated from it, control the access of different users is costly and it produces too much overhead.

The proposed encoding is depicted in Fig. 2 by means of an example. At the end of each *signal register* there is a tail, composed of two bytes, which points at the beginning of the *secure frame* to allow its retrieval. The *secure frame* is composed of:

- a *recovery container* (RC, mandatory in the first *signal register*), which includes the syntax necessary to make the content of each data *container* retrievable to targeted users (or to everybody);

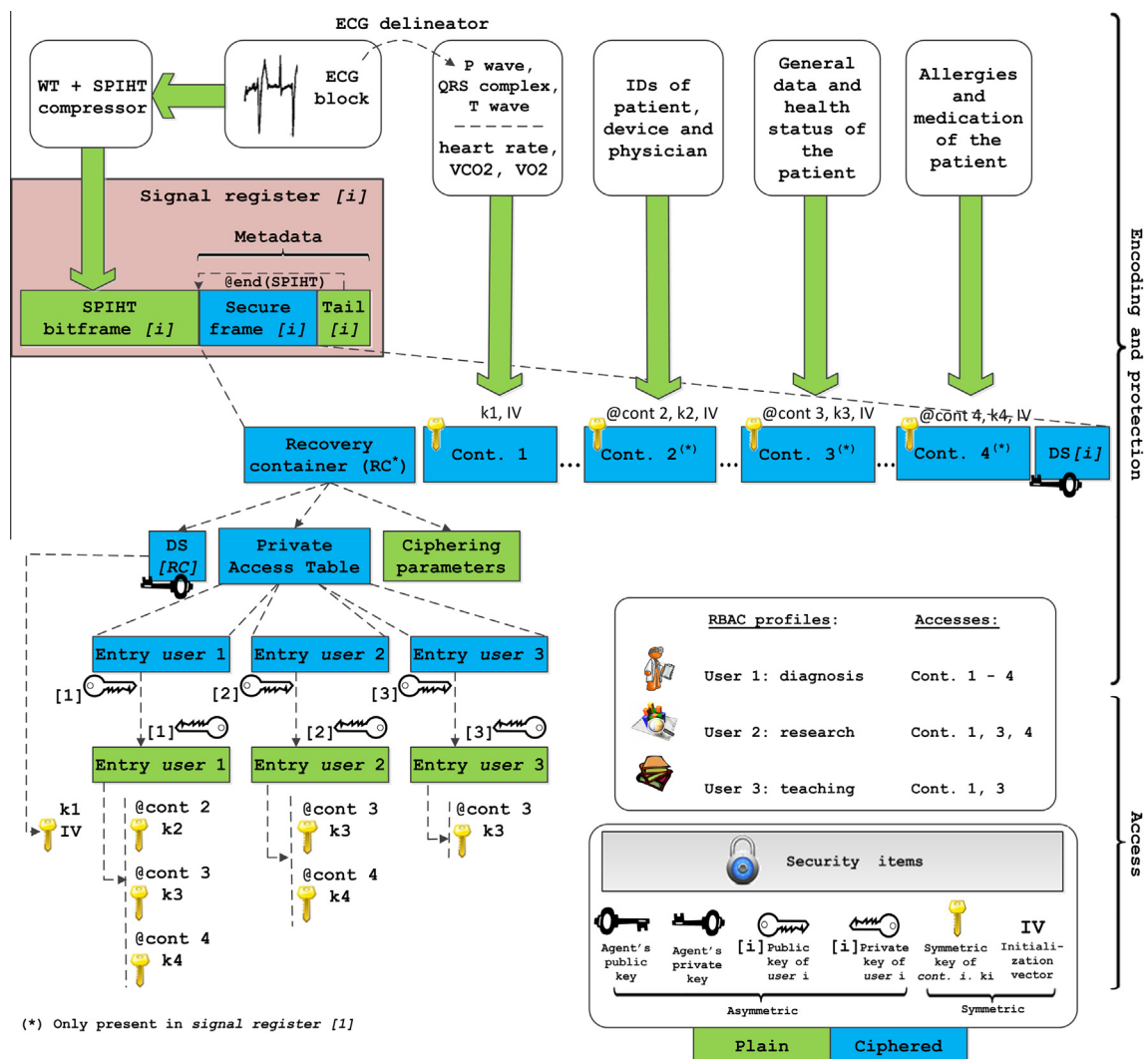


Fig. 2. Structure and content of a *signal register*. Additional measures and data of the patient are put into separated *containers* (1–4) and ciphered. The RBAC profiles establish the access data to be included in each user's entry of the Private Access Table. By using his/her private key, a user deciphers his/her access data and retrieves the *containers* authorized by his/her RBAC profile. Digital Signatures (DS) are used for authentication and checking the integrity of the RC and the *signal registers*.

- data *containers* (1–6, optional), which include ciphered context-related metadata about the test (e.g. *cont. 1*: ECG delineation, *cont. 2*: allergies, *cont. 3*: patient ID, *cont. 4*: general data and health status of the patient); and
- a Digital Signature (DS, mandatory), which allows the detection of tampering.

2.3.1. Protection scheme

The **recovery container**, depicted in Table 1, details the symmetric (*tag 3*) and asymmetric elements (*tag 0–2*) combined in the protection scheme to obtain an optimum tradeoff security-performance. *Tags 4–5* indicate the position of *container 1* to allow public access and *tag 6* contains the Private Access Table, which regulates the private access to the rest of *containers* (2–6).

The **data containers**, as illustrated in Fig. 2, are ciphered independently with symmetric cryptography, which operates very fast. A secret key (*ki*) and a initialization vector (*IV*) are used for ciphering and deciphering, which provides confidentiality. The symmetric cipherer (Table 1: *tag 3* – AES [36], Blowfish, RC6, Twofish or 3DES) operates in Output Feedback Mode (OFB), which makes cryptanalysis more difficult and does not require extra bytes for padding. The preferred cipherer is the standard AES, expected to remain secure beyond 2030 (according to NIST [33]) and faster than the rest in generation of keys and ciphering [34]. Asymmetric cryptography, which is safer and does not need previous key arrangements to start, is used to protect the symmetric key, position and length of each *container* (Table 1: *tag 6*), obtaining confidentiality and authentication. This uses a key pair (public key, private key) for ciphering and deciphering. Regarding algorithms, only the widespread RSA [37] is permitted, since its major competitor, ElGamal, is not a standard and encrypts more slowly. The access procedure is as follows:

- For the non-confidential **container 1**, its symmetric key (*k1*) and initialization vector (*IV*) are obtained from a public security element, the Digital Signature (DS) of the RC (Table 1: *tag 2*). The key corresponds to the first bits (e.g. for AES, from 1 to 256) and the *IV* to the following (e.g. for AES, from 257 to 384). The length (Table 1: *tag 4*) and position of this *container* are also public, it begins just after the RC in the first *signal register* and after the SPIHT bits in the remaining (Table 1: *tag 5*). Thus, this *container* can be accessed by anyone knowing this encoding.
- For the confidential **containers 2–6**, their symmetric keys (*ki*) and locations (positions and lengths) make private entries in the Private Access Table (PAT, Table 1: *tag 6*), encrypted with the public RSA key of the intended *user/s*. Each *user* decodes his/her entry in the PAT using his/her private key. The *IV* is the same as for *container 1*. There are noisy bytes preceding the *containers* to increase the cost of an attack due to the secrecy of their locations.

The **Digital Signatures (DS)** included in each *signal register* (Fig. 2) and in the RC (Table 1: *tag 2*) are used to check their integrity and authenticity. The DS are calculated by combining a safe hash function (Table 1: *tag 1* – SHA2 512, SHA 1 or RIPEMD 160) which makes a digest of the RC/*signal register*, with a public-key algorithm, which encrypts the digest with the private key of the *agent* who protects the test (a person, a program or the signal acquisition device itself). At *user's* side, each DS is verified by calculating the hash of the received RC/*signal register* and comparing it with the original hash, decrypted with the public key of the *agent* (extracted from his/her digital certificate, Table 1: *tag 0*). If they match the RC/*signal register* is valid, otherwise all *signal registers*/that *signal register* are refused. Regarding algorithms, RSA [37] is not allowed since the signatures are lengthy, DSA [38] is permitted

since it is very extended and its signature is very compact (see Section 3.3), and ECDSA [39] is the preferred option since the signature length is the same as DSA and its key, and consequently its digital certificate, is much shorter (see Table 4).

2.3.2. Role-Based Access Control (RBAC)

Since biomedical tests may be requested for different uses (e.g. diagnosis, research, teaching), the implementation of a RBAC policy defining different access profiles is a smart way to fulfill the privacy principles of necessity of data processing and purpose binding. The *agent* (person, program or acquisition device) will assign a RBAC profile to each intended *user*, according to his/her professional role, to establish the contents of the test that he/she is allowed to access. These policies have gained attention in recent years and currently they are integrated in several medical standards (e.g. DICOM [40] and HL7 [41]).

As illustrated in Fig. 2, the proposed RBAC policy is defined on top of the formerly described *container*-level encryption, which already allowed different *users* to access different contents (placed into separated *containers*) of a *signal register*. A possible definition for the *containers*, integrating the most interesting contents included by major medical standards (DICOM [22], HL7 [42], SCP-ECG [43] and MFER [44]), would be:

- **Container 1.** This may include information concerning the acquisition session:
 - context-aware data (e.g. type of test: resting ECG, stress ECG, ambulatory ECG monitoring, intensive care monitoring);
 - environmental parameters (e.g. positioning, humidity, temperature);
 - parameters of the signal (e.g. sampling frequency, quantization bits, amplitude multiplier, applied filters);
 - additional data extracted after signal processing (e.g. delineation of fiducial points in an ECG record, intervals of likely seizure in EEGs);
 - periodic measures acquired in intensive care monitoring (e.g. non-invasive blood pressure – NiBP, temperature – Temp, blood oxygen saturation – SPO2, carbon dioxide – CO2, heart rate);
 - periodic measures acquired in stress tests (e.g. maximal oxygen consumption – VO2, heart rate, concentration of lactate in the blood, carbon dioxide production – VCO2, speed of the treadmill/power of the bicycle).
- **Container 2.** This may include the identification of the patient (e.g. name, surname, Social Security Number, Personal Health Record identifier), the physician/technician who acquires the signal, the acquiring and analyzing devices and the institution (and/or department) that leads the test.
- **Container 3.** This may include general data (e.g. age, height, weight) and health status of the patient (e.g. diseases, symptoms, previous diagnoses, observations).
- **Container 4.** This may include the allergies and current medication of the patient.
- **Container 5.** This may include *sensitive* diseases of the patient (e.g. AIDS, venereal diseases), not included in *container 3* for confidentiality reasons.
- **Container 6.** This may include billing information of the medical test.

The corresponding RBAC profiles were defined after consulting medical experts and they aim at covering the spectrum of applications for tests in the e-health context:

0. Emergency care/surgery: access to signals, all personal and medical data of the patient, *containers 1–5*.

Table 1Structure and content of the *recovery container* (RC). ([35]See below-mentioned references for further information.)

Tag ¹	Length ¹	Value (parameter data)																																											
0	length ²	Certificate of the <i>agent</i> (PEM encoding): The agent is the person, software or acquiring device that generates the <i>signal register</i> . The certificate must be X.509, any version. Two signature algorithms are allowed: DSA [38] and ECDSA [39].																																											
		<table><tr><td></td><td>Value</td><td>Algorithm</td></tr><tr><td rowspan="3">Hash function ID</td><td>0</td><td>SHA2 512</td></tr><tr><td>1</td><td>SHA1</td></tr><tr><td>2</td><td>RIPEMD 160</td></tr></table> It is recommended to use SHA2 512.		Value	Algorithm	Hash function ID	0	SHA2 512	1	SHA1	2	RIPEMD 160																																	
	Value	Algorithm																																											
Hash function ID	0	SHA2 512																																											
	1	SHA1																																											
	2	RIPEMD 160																																											
2	length ²	Digital signature , DS = Enc(PrKa, hash(<i>RC</i>)) This is the encryption of the hash of the <i>RC</i> using the private key of the <i>agent</i> (initially DS = blank). At <i>user</i> 's side the DS is used to verify the integrity of the <i>RC</i> and authenticate the <i>agent</i> . The length depends on the signature algorithm: DSA or ECDSA.																																											
3	1 byte	<table><tr><td></td><td>Value</td><td>Algorithm</td></tr><tr><td rowspan="5">Ciphering algorithm ID</td><td>0</td><td>Rjindael (AES)</td></tr><tr><td>1</td><td>Twofish</td></tr><tr><td>2</td><td>RC6</td></tr><tr><td>3</td><td>Blowfish</td></tr><tr><td>4</td><td>3DES</td></tr></table> Rjindael is the preferred option: it was chosen as the Advanced Encryption Standard by the National Institute of Standards and Technology (NIST) after a contest [35]; Twofish and RC6 being finalists. Blowfish is a symmetric block cipher that can be used as a drop-in replacement for DES or IDEA. Although 3DES has been replaced by AES, it is sometimes used for e-commerce.		Value	Algorithm	Ciphering algorithm ID	0	Rjindael (AES)	1	Twofish	2	RC6	3	Blowfish	4	3DES																													
	Value	Algorithm																																											
Ciphering algorithm ID	0	Rjindael (AES)																																											
	1	Twofish																																											
	2	RC6																																											
	3	Blowfish																																											
	4	3DES																																											
4	2 bytes	Length of <i>container</i> 1 (bytes) in the first <i>signal register</i> .																																											
5	3 bytes	Last block including <i>container</i> 1																																											
6	length ²	<table><tr><td></td><td>Length¹</td><td>Content</td><td>Length¹</td><td>Content</td></tr><tr><td></td><td>(bytes)</td><td></td><td>(bytes)</td><td></td></tr><tr><td rowspan="7">Private Access Table (PAT)</td><td>8</td><td>Date of protection</td><td>6</td><td><i>User</i>'s public key (first bytes)</td></tr><tr><td></td><td>(seconds since January 1,</td><td>1</td><td>RBAC profile (n)</td></tr><tr><td></td><td>1970, 00:00:00 GMT)</td><td>4</td><td>Position of <i>container</i> 2 (bytes)</td></tr><tr><td>length</td><td>Enc(PbKu1, entry for <i>user</i> 1)</td><td>3</td><td>Length of <i>container</i> 2 (bytes)</td></tr><tr><td>length</td><td>Enc(PbKu2, entry for <i>user</i> 2)</td><td>length</td><td>Secret key for <i>cont.</i> 2</td></tr><tr><td>...</td><td>...</td><td>...</td><td>...</td></tr><tr><td>length</td><td>Enc(PbKuk, entry for <i>user</i> <i>k</i>)</td><td>4</td><td>Position of <i>container</i> 6 (bytes)</td></tr></table> each entry is <table><tr><td>3</td><td>Length of <i>container</i> 6 (bytes)</td></tr><tr><td>length</td><td>Secret key for <i>cont.</i> 6</td></tr></table>		Length ¹	Content	Length ¹	Content		(bytes)		(bytes)		Private Access Table (PAT)	8	Date of protection	6	<i>User</i> 's public key (first bytes)		(seconds since January 1,	1	RBAC profile (n)		1970, 00:00:00 GMT)	4	Position of <i>container</i> 2 (bytes)	length	Enc(PbKu1, entry for <i>user</i> 1)	3	Length of <i>container</i> 2 (bytes)	length	Enc(PbKu2, entry for <i>user</i> 2)	length	Secret key for <i>cont.</i> 2	length	Enc(PbKuk, entry for <i>user</i> <i>k</i>)	4	Position of <i>container</i> 6 (bytes)	3	Length of <i>container</i> 6 (bytes)	length	Secret key for <i>cont.</i> 6
			Length ¹	Content	Length ¹	Content																																							
	(bytes)		(bytes)																																										
Private Access Table (PAT)	8	Date of protection	6	<i>User</i> 's public key (first bytes)																																									
		(seconds since January 1,	1	RBAC profile (n)																																									
		1970, 00:00:00 GMT)	4	Position of <i>container</i> 2 (bytes)																																									
	length	Enc(PbKu1, entry for <i>user</i> 1)	3	Length of <i>container</i> 2 (bytes)																																									
	length	Enc(PbKu2, entry for <i>user</i> 2)	length	Secret key for <i>cont.</i> 2																																									
																																									
	length	Enc(PbKuk, entry for <i>user</i> <i>k</i>)	4	Position of <i>container</i> 6 (bytes)																																									
3	Length of <i>container</i> 6 (bytes)																																												
length	Secret key for <i>cont.</i> 6																																												
		The PAT sets accurately the test protection date by means of a NIST Internet Time Server. This establishes a single time reference and prevents attempts of forgery, but also requires an Internet connection. The PAT also allows customized multiuser access to be granted, by defining an entry per <i>user</i> , encrypted with his/her RSA [37] public key (except the first field, which identifies the user). The field <i>position of container i</i> is the distance between the end of the SPIHT bitframe and the beginning of <i>container i</i> .																																											

¹ The fields Tag and Length are represented with 1 and 2 bytes respectively.² The length of these fields is specified in Tab. 4.

1. Diagnose (by the physician who interprets the test): access to signals, all personal and medical data, excluding *sensitive* diseases not related to the current test, *containers 1–4*.
2. Research or examination (by another physician caring for the patient): access to signals and medical data of the patient preserving his/her anonymity, *containers 1, 3, 4*.
3. Teaching: access to signals and general health status of the patient to enable correlations, *containers 1, 3*.
4. Billing: access to signals and information about the cost of the acquisition session, *containers 1, 6*.
5. Signals consultation: access to signals only and *container 1*. This profile is public, available to everyone.

Nonetheless, the proposed encoding can work with different number and alternative definitions of *containers* and RBAC profiles, it is not specifically intended for these examples only.

3. Evaluation

The methods selected for signal compression (Section 2.1), metadata embedding (Section 2.2) and protection (Section 2.3) depend on several parameters (e.g. signal distortion threshold, signal block length, wavelet decomposition level, DS type) which must be studied and set up to ensure (a) user satisfaction: signal fidelity, low delays (to allow real-time operation), ease of use of the implementation (see Section 4) and (b) optimal system features: low bandwidth requirements, low overhead of the security elements and enough embedding capacity to include data produced in e-health services.

A variety of electrocardiograms (ECGs), commonly used for the detection and diagnosis of heart disease, and electroencephalograms (EEGs), relevant in applications such as brain-computer interfaces and the study of epilepsy and sleep disorders (insomnia, circadian rhythm disorders, parasomnia, etc.) are used to carry out all the parts of this evaluation.

3.1. Databases

Two well-known ECG databases have been used. The first one is the Massachusetts Institute of Technology (MIT)–Beth Israel Hospital (BIH) Arrhythmia [45]. This ECG database consists of 48 two-lead ECG registers of 30 min duration. The sampling rate is 360 samples per second with a resolution of 11 bits per sample. Although the database was originally created as standard test material for the evaluation of arrhythmia detectors, this database is by far the most used to test and compare ECG compression algorithms. The second ECG database is MIT-BIH Compression [45]. It is composed of 168 two-lead ECG records of 20.48 s duration. The sampling rate is 250 samples/s with a resolution of 12 bits per sample. This database was created to pose a variety of challenges for ECG compressors, in particular for lossy compression methods. Despite this fact, it is scarcely used to test the ECG compression algorithms, being relegated by MIT-BIH arrhythmia. Since these ECG databases are composed of two-lead recordings, the entire evaluation was run on both leads and the results represent the average.

For testing with EEGs we chose the STUDY dataset [46,47] from the Swartz Center for Computational Neuroscience (SCCN), composed of 10 recordings from 5 different subjects, with 61 channels per frame, 820 frames per epoch and 220–235 epochs. The sampling rate of these recordings is 200 samples/s and the resolution is 11 bits per sample.

3.2. Bounding signal distortion

Fidelity of a compressed signal is understood as the close similarity with respect to the original. In clinical applications, it is

essential to measure the distance between both signals by means of some distortion measure and setting a quality threshold to regulate the compression process. Among the most widespread measures of signal distortion are:

- the *Root Mean Square error (RMS)*, defined as

$$RMS = \sqrt{\frac{(\chi(n) - \tilde{\chi}(n))^2}{N}}, \quad \text{and} \quad (1)$$

- the *Percentage RMS Distortion (PRD)*, defined as

$$PRD = \sqrt{\frac{\sum_{n=1}^N (\chi(n) - \tilde{\chi}(n))^2}{\sum_{n=1}^N (\chi(n) - \bar{\chi})^2}} \cdot 100, \quad (2)$$

where $\chi(n)$ is the original signal, $\tilde{\chi}(n)$ is the reconstructed, $\bar{\chi}$ the mean of the original signal and N its length.

It can be observed in Eq. (1) how the amplitude range of the signal affects the measure: compressed signals constrained to lower amplitudes obtain lower *RMS* than those with higher amplitude and the same fidelity. The definition of *PRD* in Eq. (2) overcomes this issue because it uses a normalization which is independent from the amplitude of the signal (and from its DC level). Thus, our choice is using the *PRD* as the measure of signal distortion, since it allows much fairer comparisons.

Furthermore, the correlation between the *PRD* and the mean opinion score (MOS) of expert cardiologists, obtained through blind and semi blind tests, was studied by Zigel in [48]. One of the conclusions of that work was that all tested signals with *PRD* < 9% were considered as “good” or “very good” by the cardiologists. Thus, this value is used as quality threshold for ECG compression in our system. Similarly, other works relate *PRD* to EEG quality. In [49] it is suggested limiting *PRD* to 7% to maintain 99.5% of the signal energy, while in [50] it is proposed rising to 30% since this value allows a seizure detection rate of 90% to be reached in epilepsy monitoring (using REACT, a state-of-the-art algorithm). Among these two values, 7% is preferred since EEG records may be used in applications requiring higher quality than seizure detection. Fig. 3 shows two signals from our databases, an ECG and an EEG, which are compressed with the proposed thresholds and retain their main shapes accurately.

3.3. System delays and bandwidth requirements

The delays of the proposed encoding and access system are estimated in Table 2. The overall delay is mainly contributed by the latency of acquiring a signal block, expressible as:

$$\text{block length}(s) = \frac{\text{block length} (\# \text{samples})}{\text{sampling freq.} (\# \text{samples/s})}, \quad (3)$$

In fact, in all the configurations presented this delay is much greater than the sum of delays of the remaining processes (see Table 2: sub-total). This enables real-time operation on the condition of signal blocks as short as possible to maintain the delay at acceptable levels.

The bandwidth required for the transmission of encoded ECG and EEG signals when embedding security elements (mandatory) and additional metadata (optional) is evaluated in Table 3. Four observations were made. First, using long signal blocks produces a decrease in signal bandwidth requirements which stops at 4096 samples/block for ECGs, in the case of EEGs higher values can improve the compression at the cost of very high delays (>20.48 s according to Eq. (3)). Long signal blocks allow more signal cycles to be included in a single block, lower frequencies obtain higher relevance and this benefits the sorting of the temporal

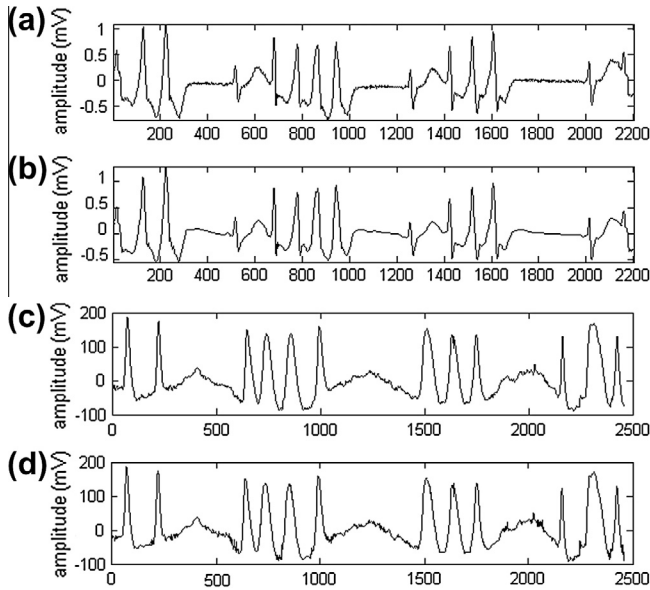


Fig. 3. Signals (a) 08730_2 (lead 1) ECG from MIT-BIH Compression (bitrate 3000 bps), (b) compressed with PRD = 9% (bitrate 202 bps), (c) Syn08-s254 EEG from SSCN (bitrate 2200 bps), and (d) compressed with PRD = 7% (bitrate 240 bps). Additional parameters: block length = 512 samples, wavelet decomposition level = 6.

orientation trees used by the 1D SPIHT, which increases the efficiency of the compression. Second, the signal bandwidth requirements increase slightly ($\leq 4\%$) when embedding a big amount of metadata (three last columns in Table 3). Nevertheless this only happens in the first *signal register*, since the rest do not include *containers* 2–6. Third, using long signal blocks dramatically reduces the metadata bandwidth requirements since, in each *signal register*, the size of the security elements with respect to the size of the encoded signal block is lower. Fourth, the encoded signal compresses the original (compression rate > 1) despite embedding security elements and metadata. The only exception appears in the first *signal register* when using short signal blocks (512 samples/block) and embedding more than 3 KB in *containers* 2–6.

The size of the contents arranged in *secure frames* and subsequently embedded within *signal registers* for the bandwidth evaluation above are depicted in Table 4. The chosen agent's certificate type for DS was ECDSA 239 and we considered the case of 5 users with RSA 1024 certificates when embedding *containers* 2–6. For ECGs, the *container* 1 included the signal delineation and additional measures obtained from a stress test (VO₂, heart rate, concentration of lactate in the blood, VCO₂ and speed of the treadmill). For EEGs, the *container* 1 included a likelihood measure for EEG seizure detection and additional monitoring measures (NiBP, Temp, SPO₂, CO₂ and heart rate). The ECG delineation consisted of the position of 15 fiducial points (wave onsets, peaks and offsets) per cardiac cycle, each point encoded with 2 bytes. EEG seizure detection likelihood was estimated every second and encoded using 1 byte. Each additional measure was recorded at 1 sample/s and encoded using 1 byte. For *containers* 2–6 we estimated that its overall size is around 3–10 KB. Although they store a lot of different medical data (see Section 2.3:RBAC), most of it can be described by means of IDs.

3.3.1. Parameters tuning

As demonstrated above, the length of the signal block establishes a tradeoff between the system overall delay and the bandwidth required for the transmission. Therefore two different values are recommended according to the application.

- 512 samples/block for **real-time transmission**, which yields acceptable delays (see Table 2: total) and low signal transmission rates (see Table 3): MIT-Arrhythmia (2 s, 409 bps/lead), MIT-Compression (2.7 s, 309 bps/lead), SSCN-EEG (3.3 s, 474 bps/channel).
- 4096 samples/block for **offline transmission**, which produces longer delays but more efficient signal transmission: MIT-Arrhythmia (12 s, 373 bps/lead), MIT-Compression (17 s, 282 bps/lead), SSCN-EEG (22.3 s, 389 bps/channel). Besides the metadata transmission rate is reduced to one eighth with this configuration.

The signal compression, described in Section 2.1, begins with the wavelet transformation of the signal block. The Coiflet filter with 12 coefficients was chosen for this transformation, since it obtains higher compression efficiency than others (e.g. Daubechies with 20 coefficients) and offers a good tradeoff between the number of operations and the quality of the reconstructed signal. The wavelet decomposition level was set to 6 because we observed that the compression efficiency improves notably until this level but not in the following.

The protection scheme, described in Section 2.3, introduces overhead due to the need of including a Digital Signature (DS) in each *signal register* (see Fig. 2). Several signature algorithms provide similar security with different DS length: DSA [38] (1024, 2048, 4096) and ECDSA [39] (192, 224, 339) generate signatures sized in the range [0.05,0.06] KB, while RSA [37] (1024, 2048, 4096) signatures result much longer [0.13, 0.26, 0.51] KB. To reduce the overhead, only DSA and ECDSA signatures are allowed.

3.4. Embedding capacity

We define *embedding capacity* (EC) as the amount of metadata that can be embedded with our encoding method when using the same bandwidth as for transmitting the signal uncompressed. The EC_i (per lead/channel) of different ECG and EEG signals are shown in Table 5. In most cases the overall EC (e.g. ≥ 77.7 MB in ambulatory recordings –25.9 MB · 3 leads– or ≥ 2.15 MB in stress tests –178.9 KB · 12 leads) far exceeds the size that *containers* 1–6 require to enable e-health services, estimated in Table 4. The difference is what it is saved in transmission and storage, typically ≈ 70 –80% of the original size.

Each *signal register* j from a lead/channel i has its own embedding capacity, EC_{ij} (depicted in Fig. 4a), resulting from the difference between the sizes of the original signal block and the corresponding *signal register* (SPIHT bitframe, *secure frame* and tail). The $EC_i(t)$ of a lead/channel i , illustrated in Fig. 4b, is the sum of the EC_{ij} of the blocks 1 to j transmitted/stored until t . The size of the RC, embedded in the first *signal register*, corresponds to the negative offset in Fig. 4b. The embedding capacity of a lead/channel can be approximated as:

$$EC_i(t) = (\text{sampling freq} \cdot \text{bit res} - \text{compressed signal bitrate} - \frac{\text{size(DS)}}{\text{block length}}) \cdot t - \text{size(RC)}. \quad (4)$$

To build Table 5, we used this approximation. The sampling frequencies of the signals and their resolutions were consulted in Section 3.1, the compressed signal bitrates in Table 3, the size of the DS (we considered ECDSA 239) in Table 4 and the block length was calculated with Eq. (3).

4. Implementation and use

The implementation of the encoding and access architecture, depicted in Figs. 1 and 2, is openly available at

Table 2
Average delay of the system operations.

Parameters			Delay (ms)										Block length(t), see Eq. (3)	Total
Database	Block length	PRD (%)	Del./ seiz. det.	Cmp.	Cont.- level encr.	DS	Tr.	DS check	Dec.	RBAC access	Sub total			
Arrhythmia (ECG)	512	9	13	0.2	360	30	0.2	30	0.1	180	613.2	1422	2035.2	
Compression (ECG)	512	9	13	0.3	360	30	0.2	30	0.1	180	613.2	2048	2661.2	
SCCN (EEG)	512	7	156	0.3	360	30	0.4	30	0.1	180	756.4	2560	3316.4	
Arrhythmia (ECG)	4096	9	37	2.4	360	30	1.3	30	1.2	180	638.3	11,380	12018.3	
Compression (ECG)	4096	9	37	3.2	360	30	1.8	30	1.5	180	638.8	16,380	17018.8	
SCCN (EEG)	4096	7	1248	3.0	360	30	3.1	30	1.4	180	1851.1	20,480	22331.1	

Abbreviations: *del./seiz. det.* is ECG delineation/ EEG seizure detection, *comp.* is SPIHT compression, *cont.-level encr.* is container-level encryption, *DS* is calculating the digital signature of the *signal register*, *tr.* is transmitting the *signal register* using HSUPA at 5.76 Mbps, *DS check* is checking the digital signature, *dec.* is SPIHT decompression, *RBAC access* is decrypting the *containers* allowed to the intended user.

Table 3
Encoded signal bitrate to maintain constant distortion after embedding secure frames (SF) with different elements.

Parameters			Average compressed signal bitrate -bps/lead- +metadata bitrate -bps/lead- (overall compression ratio)					
Database	Block length	PRD (%)	No SF	DS only	Cont. 1 & DS	RC, cont. 1, cont. 2–6 (3 KB) & DS	RC, cont. 1, cont. 2–6 (6 KB) & DS	RC, cont. 1, cont. 2–6 (10 KB) & DS
Arrhythmia (ECG)	512	9	409.4 (9.67)	409.7 + 368.9 (5.09)	410.6 + 648.6 (3.74)	422.0 + 2809 (1.23)	422.5 + 4969 (0.73)	422.2 + 7849 (0.48)
Compression (ECG)	512	9	309.2 (9.70)	307.7 + 256.0 (5.32)	308.2 + 536.0 (3.55)	317.1 + 2036 (1.27)	316.8 + 3536 (0.78)	317.0 + 5536 (0.51)
SCCN (EEG)	512	7	474.2 (4.64)	459.3 + 204.8 (3.31)	459.2 + 252.8 (3.09)	473.1 + 1453 (1.14)	472.7 + 2653 (0.70)	472.8 + 4253 (0.47)
Arrhythmia (ECG)	4096	9	372.5 (10.63)	372.7 + 46.1 (9.46)	372.8 + 326.1 (5.67)	387.7 + 596.1 (4.03)	387.6 + 866.1 (3.16)	387.5 + 1226 (2.45)
Compression (ECG)	4096	9	282.0 (10.64)	282.1 + 32.0 (9.55)	282.3 + 312.0 (5.05)	290.5 + 499.5 (3.80)	290.5 + 687.0 (3.07)	290.5 + 937.0 (2.44)
SCCN (EEG)	4096	7	388.6 (5.66)	386.6 + 25.6 (5.34)	386.3 + 73.6 (4.78)	395.2 + 223.6 (3.56)	394.8 + 373.6 (2.86)	394.9 + 573.6 (2.27)

<http://sourceforge.net/projects/pfmt/>. It is divided into three modules: a standard 1D SPIHT coder-decoder (see Section 2.1), whose optimal parameters of (real-time/offline) operation were studied in Section 3.3; a Graphical User Interface (GUI) to build/access the *secure frames* and some simple codes to embed the *secure frames* and retrieve them from the *signal registers* (see Section 2.2).

As illustrated in Fig. 5, the design of the GUI is rather simple and intuitive, to encourage the use even among *users* with little technical knowledge. It facilitates the encoding of additional measures and data of the patient in the corresponding *containers*, the assignment of role-based access profiles to intended *users* (physicians, researchers, teachers, etc.) and the protection of the resulting *signal registers*. All the corresponding operations of ciphering, decipher-

ing, signing and checking are carried out by the GUI. However, some interaction is required:

- With the *agent* who encodes the *signal registers*, he/she must:
 1. load the SPIHT bitframes;
 2. load the content of the data *container/s* (1–6), some may be empty;
 3. load the certificates of the *users* and indicate their RBAC profiles (0–5);
 4. load his/her digital certificate, if desired change the default hash function and the ciphering algorithm;
 5. load his/her password-protected private key;
 6. save the resulting *signal registers*.

Table 4
Typical size (KB) of the containers in a secure frame.

Recovery container (RC), defined in Table 1			Rest of containers			
Agent's cert	RC-Tag 0	RC-Tag 2	signal register DS	Database	Block length	Container 1
ECDSA 192	0.6	0.05	0.05	Arrhythmia (ECG)	512	0.053
ECDSA 224	0.6	0.06	0.06	Compression (ECG)	512	0.076
ECDSA 239	0.6	0.06	0.06	SCCN (EEG)	512	0.012
DSA 1024	1.1	0.05	0.05	Arrhythmia (ECG)	4096	0.421
DSA 2048	1.6	0.05	0.05	Compression (ECG)	4096	0.606
DSA 4096	2.6	0.05	0.05	SCCN (EEG)	4096	0.098
RC-Tags 1, 3–5	User's cert	RC-Tag 6	Containers 2–6			
0.007	RSA 1024	0.13 · #users	3–10			
	RSA 2048	0.26 · #users				
	RSA 4096	0.51 · #users				

Table 5Average embedding capacity (EC_i) per lead/channel of different ECGs and EEGs.

Test and duration	Signal database	Samples/block	EC_i
Resting ECG, 10 s	Arrhythmia	512	66.5% (3.2 KB)
	Arrhythmia	4096	75.6% (3.7 KB)
	Compression	512	63.0% (2.3 KB)
	Compression	4096	71.3% (2.6 KB)
Resting ECG, 30 s	Arrhythmia	512	75.7% (11.0 KB)
	Arrhythmia	4096	84.8% (12.3 KB)
	Compression	512	75.1% (8.3 KB)
	Compression	4096	83.5% (9.2 KB)
Stress ECG, 10 min	Arrhythmia	512	80.1% (232.4 KB)
	Arrhythmia	4096	89.2% (258.7 KB)
	Compression	512	80.9% (177.8 KB)
	Compression	4096	89.2% (196.1 KB)
Ambulatory ECG, 24 h	Arrhythmia	512	80.3% (33.6 MB)
	Arrhythmia	4096	89.4% (37.4 MB)
	Compression	512	81.2% (25.7 MB)
	Compression	4096	89.5% (28.3 MB)
Epilepsy detection (EEG), 30 min	SCCN-EEG	512	69.7% (336.8 KB)
	SCCN-EEG	4096	81.1% (392.2 KB)
Polysomnographic study (EEG), 6.5 h	SCCN-EEG	512	69.8% (4.4 MB)
	SCCN-EEG	4096	81.3% (5.1 MB)

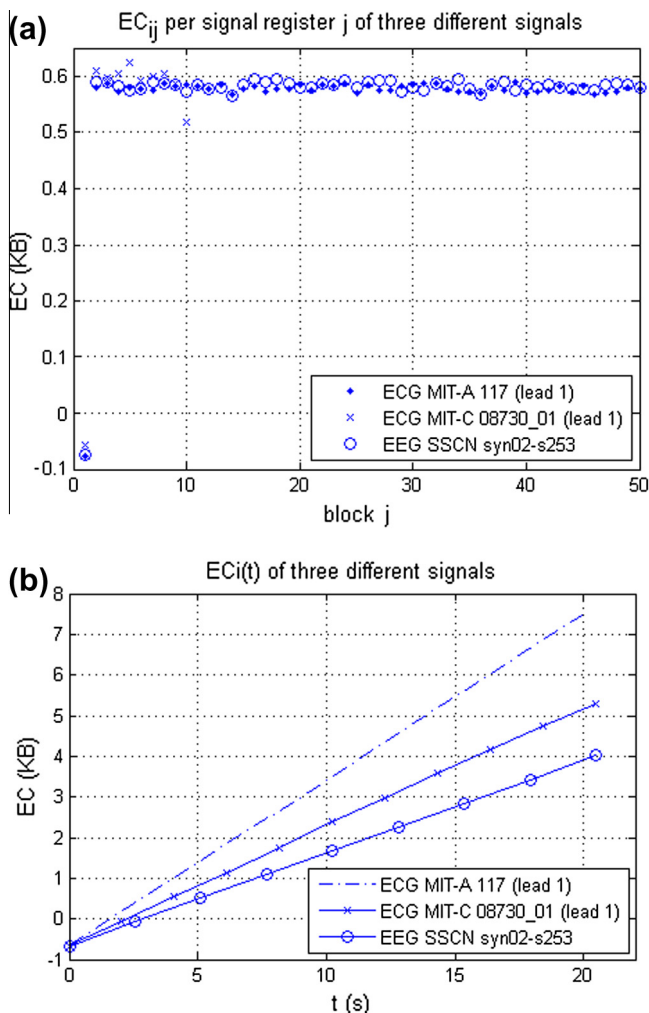


Fig. 4. Embedding capacity (EC) per ECG register (a) and per lead (b) of two ECGs from MIT-Arrhythmia and MIT-Compression and an EEG from SSCN-EEG. ECGs compressed with $PRD = 9\%$, EEG with $PRD = 7\%$, block length = 512 samples, wavelet decomposition level = 6.

- With the *user* who accesses the *signal registers*, he/she must:
 1. load the *signal registers*;
 2. export the Private Access Table to check his/her RBAC profile (if desired);
 3. load his/her password-protected private key (only if he/she is allowed to access *containers* 2–6);
 4. save the *container/s* that he/she is allowed to access.

These interactions with *agents* and *users* could be minimized by defining system configuration profiles. Due to the cryptographic operations involved, it is necessary that each *agent* and each *user* possess his/her own digital certificate (and the coupled password-protected private key). However, this requirement did not decrease the experience of the consulted physicians, who pointed out that the GUI was easy to handle. The certificates associated with electronic IDs are valid for this purpose.

5. Conclusions

The proposed encoding and access architecture for 1D biomedical signals looks for user satisfaction, since it guarantees clinical value of signals, permits real-time operation (overall delays about 2–3.3 s) and can be easily handled by PACS and users through an intuitive interface that we provide. Besides, the system is very efficient and secure. It permits embedding large amounts of additional information within the signal (e.g. ≈ 3 KB per lead in resting ECGs, ≈ 200 KB per lead in stress tests, ≈ 30 MB per lead in ambulatory ECGs), detecting corruption of the signal or the information and implementing different access levels for a variety of professional *roles*. The compression ratio achieved by the encoding is quite high, ranging from ≈ 3 in real-time transmission to ≈ 5 in offline operation, despite of the embedding of security elements and metadata to enable e-health services.

In addition, this architecture could be extended to biomedical signals of higher dimension (e.g. 2D: MRIs, TACs, 3D: echocardiograms), since the encoding relies on the SPIHT algorithm. This would only require changing to the adequate SPIHT modality (2D/3D), tuning the compression parameters and establishing the distortion threshold for each new type of signal. Regarding

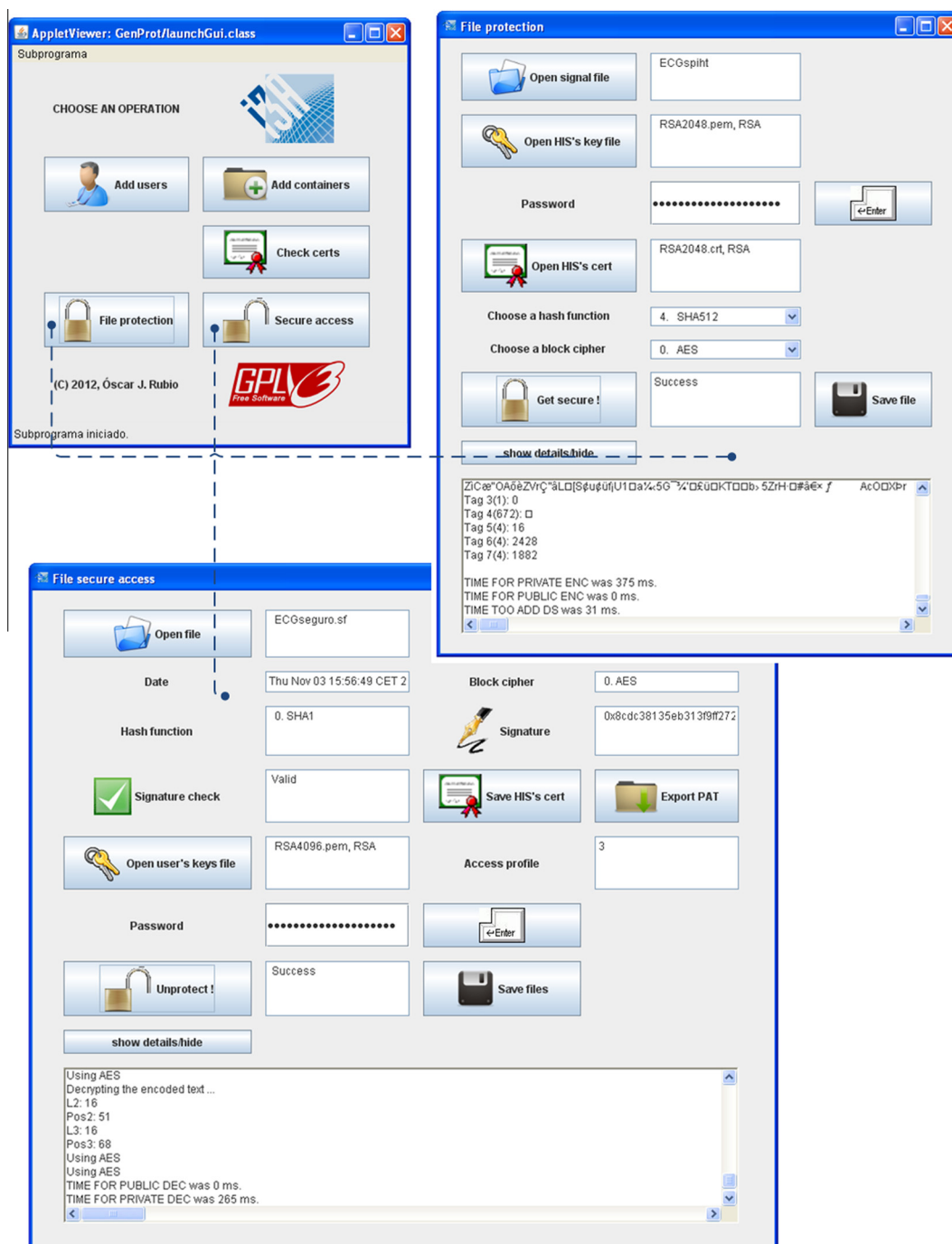


Fig. 5. Graphical User Interface to build and access secure frames, depicted in Fig. 2.

the protection scheme, although it is based on robust cryptographic elements we plan to add watermarks as a security supplement.

To sum up, this architecture fosters the secure and efficient storage, transmission and access to biomedical tests in healthcare environments. Furthermore, most of the system features (high signal compression with clinical quality, real-time operation, embedding within the signal, security with reduced overhead) are not currently supported by well-established signal standards (e.g. DICOM waveform 30, SCP-ECG), which makes it a promising alternative for the development of new and upgrade of existing e-health services.

Acknowledgements

This research work has been partially supported by Project TIN-2011-23792/TSI from the Ministerio de Economía y Competitividad (MINECO), the European Regional Development Fund (ERDF) and the European Social Fund (ESF).

References

- [1] Eysenbach G. What is e-health? J Med Int Res 2001;3:e20.
- [2] Ball MJ, Lillis J. E-health: transforming the physician/patient relationship. Int J Med Inform 2001;61:1–10.

- [3] Miller R, Gardner R, Johnson K, Hripscak G. Clinical decision support and electronic prescribing systems: a time for responsible thought and action. *J Am Med Assoc* 2005;12:403–9.
- [4] Jaleleddine S, Hutchens C, Strattan R, Coberly W. ECG data compression techniques – a unified approach. *IEEE Trans Biomed Eng* 1990;37:329–43.
- [5] Rao KR, Yip P, Britanak V. Discrete cosine transform: algorithms, advantages, applications. Orlando, FL, USA: Academic Press, Inc.; 2007.
- [6] Olmos S, Millan M, García J, Laguna P. ECG data compression with the Karhunen–Loève transform. *Comput Cardiol* 1996:253–6.
- [7] Akay M, Mello C. Wavelets for biomedical signal processing. In: Engineering in medicine and biology society, 1997. Proceedings of the 19th annual international conference of the IEEE, vol. 6, p. 2688–91.
- [8] Said A, Pearlman W. A new, fast, and efficient image codec based on set partitioning in hierarchical trees. *IEEE Trans Circ Syst Video Technol* 1996;6:243–50.
- [9] The Health Insurance Portability and Accountability Act (P.L.104-191). Enacted by the U.S. Congress; 1996.
- [10] The Personal Information Protection and Electronic Document Act, 2000. Enacted in Canada for protection of health information against commercial use.
- [11] Ley Orgánica de Protección de Datos de carácter personal (Organic Law for Protection of Personal Data). Enacted in Spain; 1999.
- [12] Lu Z, Kim D, Pearlman W. Wavelet compression of ECG signals by the set partitioning in hierarchical trees (SPIHT) algorithm. *IEEE Trans Biomed Eng* 2000;47:849–56.
- [13] Higgins G, McGinley B, Walsh N, Glavin M, Jones E. Lossy compression of EEG signals using SPIHT. *Electron Lett* 2011;47:1017–8.
- [14] Hyang S, Liao W. A compressed domain image watermarking scheme with the SPIHT coding. *J Inform Sci Eng* 2010;26:1755–70.
- [15] Kozat S, Vlachos M, Lucchese C, Van Herle H, Yu P. Embedding and retrieving private metadata in electrocardiograms. *J Med Syst* 2009;33:241–59.
- [16] Zheng K, Qian X. Reversible data hiding for electrocardiogram signal based on wavelet transforms. In: International conference on computational intelligence and security. CIS '08, vol. 1; 2008. p. 295–9.
- [17] Kong X, Feng R. Watermarking medical signals for telemedicine. *IEEE Trans Inform Technol Biomed* 2001;5:195–201.
- [18] Jamasebi R, Johnson NL, Kaffashi F, Redline S, Loparo KA. A watermarking algorithm for polysomnography data. In: 30th Annual international conference of the IEEE on engineering in medicine and biology society. EMBS 2008; 2008. p. 5720–3.
- [19] Engin M, Çıdam O, Engin E. Wavelet transformation based watermarking technique for human electrocardiogram (ECG). *J Med Syst* 2005;29(6):589–94.
- [20] Kaur S, Farooq O, Singhal R, Ahuja B. Digital watermarking of ECG data for secure wireless communication. In: 2010 International conference on recent trends in information, telecommunication and computing (ITC), Kochi, Kerala, India. p. 140–4.
- [21] Srinivasan K, Ramasubba Reddy M. Efficient preprocessing technique for real-time lossless EEG compression. *Electron Lett* 2010;46:26–7.
- [22] DICOM Waveform supplement 30: waveform interchange; 2000. <<http://tinyurl.com/7tlxbbt>> [accessed January 2012].
- [23] Trigo J, Chiarugi F, Alesanco A, Martínez-Espronedda M, Chronaki C, Escayola J, Martínez I, García J. Standard-compliant real-time transmission of ECGs: harmonization of ISO/IEEE 11073-PHD and SCP-ECG. In: Annual international conference of the IEEE on engineering in medicine and biology society. EMBC 2009, Minneapolis. p. 4635–8.
- [24] Trigo J, Chiarugi F, Alesanco A, Martínez-Espronedda M, Serrano L, Chronaki C, et al. Interoperability in digital electrocardiography: harmonization of ISO/IEEE x73-PHD and SCP-ECG. *IEEE Trans Inform Technol Biomed* 2010;14(6):1303–17.
- [25] Trigo JD, Alesanco A, Martínez I, García J. A review on digital ECG formats and the relationships between them. *IEEE Trans Inform Technol Biomed* 2012;16:432–44.
- [26] Rubio OJ, Alesanco A, García J. A robust and simple security extension for the medical standard SCP-ECG. *J Biomed Inform* 2013;46:142–51.
- [27] Alesanco A, García J. A simple method for guaranteeing ECG quality in real-time wavelet lossy coding. *EURASIP J Appl Signal Process* 2007;2007.
- [28] Alesanco A, García J. Automatic real-time ECG coding methodology guaranteeing signal interpretation quality. *IEEE Trans Biomed Eng* 2008;55:2519–27.
- [29] Kim B-J, Xiong Z, Pearlman W. Low bit-rate scalable video coding with 3-D set partitioning in hierarchical trees (3-D SPIHT). *IEEE Trans Circ Syst Video Technol* 2000;10:1374–87.
- [30] Ziegler G, Lensch H, Magnor M, Seidel H-P. Multi-video compression in texture space using 4D SPIHT. In: 2004 IEEE 6th workshop on multimedia signal processing. p. 39–42.
- [31] Wegmueller M, Perels D, Blaser T, Senn S, Stadelmann P, Felber N, Fichtner W. Silicon Implementation of the SPIHT algorithm for compression of ECG records. In: IEEE international midwest symposium on circuits and systems. MWSCAS '06. 49th, vol. 2; 2006. p. 381–5.
- [32] Housley R. Cryptographic message syntax (CMS), RFC 5652, IETF network working group; September 2009. <<http://tools.ietf.org/html/rfc5652>> [accessed March 2013].
- [33] Giry Damien. Cryptographic key length recommendation; February 2013. <<http://www.keylength.com/>> [accessed March 2013].
- [34] Dai Wei. Speed benchmarks for some common cryptographic algorithms, Crypto++ v5.6.0; March 2009. <<http://tinyurl.com/3uc96d>> [accessed April 2013].
- [35] Burr W. Selecting the advanced encryption standard. *IEEE Secur Privacy* 2003;1(2):43–52.
- [36] Daemen J, Rijmen V. FIPS PUB 197: advanced encryption standard (AES); November 2001. <<http://tinyurl.com/qksc6>> [accessed May 2012].
- [37] RSA Laboratories, PKCS 1: RSA cryptography standard; June 2002. <<http://tinyurl.com/blvnfv2>> [accessed May 2012].
- [38] Kravitz DW. (FIPS PUBS 186: digital signature standard (DSS); May 1994. <<http://tinyurl.com/2pxg3h>> [accessed May 2012].
- [39] Certicom Research, Standards for efficient cryptography, SEC 1: Elliptic Curve Cryptography, Version 2.0; May 2009. <<http://tinyurl.com/6sqli3d>> [accessed May 2012].
- [40] Digital imaging and communications in medicine (DICOM) Part 15: security and system management profiles. National electrical manufacturers association (NEMA), Rosslyn, VA, PS 3.15; 2011. <<http://tinyurl.com/7p8b25z>> [accessed May 2012].
- [41] HL7 Role-Based Access Control (RBAC) role engineering process; 2007. <<http://tinyurl.com/8yybvow>> [accessed May 2012].
- [42] Health Level 7 Annotated ECG, ANSI standard; 2004. <<http://tinyurl.com/7fldgw4>> [accessed January 2012].
- [43] SCP-ECG, Standard communication protocol for computer-assisted electrocardiography, ISO 11073-91064:2009; 2009. <<http://tinyurl.com/86qzff6>> [accessed January 2012].
- [44] Medical waveform description format encoding rules; 2007. <<http://tinyurl.com/88rrob3>> [accessed January 2012].
- [45] Moody G, Mark R, Goldberger A. Evaluation of the 'TRIM' ECG data compressor. In: Proceedings of computers in cardiology; 1988. p. 167–70.
- [46] Delorme A, Makeig S. EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics including independent component analysis. *J Neurosci Methods* 2004;134:9–21.
- [47] Ullsberger P, Delorme A. EEGLAB studyset; September 2007. <<http://tinyurl.com/bsdkyay>> [Accessed May 2012].
- [48] Zigel Y, Cohen A, Katz A. The weighted diagnostic distortion (WDD) measure for ECG signal compression. *IEEE Trans Biomed Eng* 2000;47:1422–30.
- [49] Cárdenas-Barrera JL, Lorenzo-Ginori JV, Rodríguez-Valdivia E. A wavelet-packets based algorithm for EEG signal compression. *Inform Health Soc Care* 2004;29:15–27.
- [50] Higgins G, Faul S, McEvoy R, McGinley B, Glavin M, Marnane W, Jones E. EEG compression using JPEG2000: How much loss is too much? In: 2010 Annual international conference of the IEEE on engineering in medicine and biology society (EMBC); 2010. p. 614–7.